­VIETNAM NATIONAL UNIVERSITY OF HOCHIMINH CITY

THE INTERNATIONAL UNIVERSITY

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING



**Exploring Makeup Removal**

**Using CycleGAN**

By

Nguyen Vu Duc Lam

Lecturer: Mai Hoang Bao An

Ho Chi Minh City, Vietnam

Year 2023

# Table of Figure

[Figure 1: Example of CNN Layer 3](#_Toc152338893)

[Figure 2 CycleGAN architecture from paper 4](#_Toc152338894)

[Figure 3 Implement Classification Model 6](#_Toc152338895)

[Figure 4 Generator Implementation 6](#_Toc152338896)

[Figure 5 Discriminator Implementation 7](#_Toc152338897)

[Figure 6 Loss Function Implementation 7](#_Toc152338898)

[Figure 7 Basic functions 8](#_Toc152338899)

[Figure 8 Example of product 9](#_Toc152338900)

[Figure 9 classification model result 9](#_Toc152338901)

[Figure 10 Gennerator loss 10](#_Toc152338902)

# Table of Table

[Table 1 Dataset summary 6](#_Toc152337292)

*Abstract**: Makup is a common part of many people’s daily routines, but this led to some problem about the trust in society: People wonder whether they should believe a picture or not, face recognition system has a challenge in dealing with makeup face. This project explores an approach to remove makeup using deep learning techniques, specifically Convolutional Neural Networks (CNN) and Generative Adversarial Networks (GANs).*

*Keyword:* *CNN, GAN, CycleGAN, remove makeup*

[*Github*](https://github.com/lam1101999/makeup-removal) *-* [*Web*](https://anti-makeup.streamlit.app/)

# Introduction

The motivation behind this project comes from the increasing demand for realistic and recoverable image tool. While makeup enhances facial features, many people abuse it to build a unrealistic profile on the Internet. There are scenarios where we cannot recognize a person because they look so different in the photo. This is also a challenge for the current facial recognition system. These systems also meet the same problem as humans, they cannot differentiate a person with a heavy makeup layer. Examples include banking security, tracking people in a large space where makeup have significant impact on the result.

Additionally, in the realm of image processing, the creation of deep learning has changed the game. These kinds of approaches force the network to learn the information itself with a limited intervention of human. It makes the process of makeup removal become possible.

# Relative Concept

## Convolutional Neural Network

A Convolutional Neural Network (CNN) is a type of neural network architecture that can take an image as input, give weights and biases (which can be learned) to different parts in the image. When compared to other classification algorithms, the amount of pre-processing required by a CNN is much lower. While primitive methods require hand-engineering of filters, CNN can learn these filters/characteristics with enough training.

**Diagram

Description automatically generated with low confidence**

Figure 1: Example of CNN Layer

## CycleGAN

The idea of creating an image from another image is becoming more and more popular today. The topic can be:

* Style-transfer [1]: Create a new image by combining the content of one image and style of other images. For example, we want to convert a comic picture into Vincent van Gogh style.
* Image-Enhancement [2] [3]: Improve the information in the current image for human viewer. It includes sub-topics: recover image, color-enhancement, remove noise, and so on.
* …

Most of them are based on the idea of Auto-Encoder and GAN [4]. Autoencoders are a specific type of feedforward neural networks where the input is the same as the output. They compress the input into a lower-dimensional code and then reconstruct the output from this representation. The code is a compact “summary” or “compression” of the input, also called the latent-space representation. An autoencoder consists of 3 components: encoder, code and decoder. The encoder compresses the input and produces the code, the decoder then reconstructs the input only using this code. GANs use a smart method to teach a generative model. They treat it like a game with two players: the generator, which makes new examples, and the discriminator, which tells real from fake. They train these players in a game until the discriminator is fooled about half the time, showing the generator makes realistic examples.

To create a makeup removal system, at first, I try 2 strategies: pix2pix [5] and Auto-Encoder. Both strategies require makeup image as input, and the target is non-makeup image. It forces models to remove makeup to reduce loss. The problem is that we need paired dataset. It means for each makeup image we need one corresponding non-makeup image of the same person. That kind of data is limited; therefore, this project tries another approach: CycleGAN [6]. They propose a method to translate an image from one type (X) to another (Y) without having paired examples. The objective is to develop a mapping, G:X→Y, in such a way that the images produced by G(X) are indistinguishable from those in Y, achieved through adversarial loss. To make it clear, this is the architecture of CycleGAN:

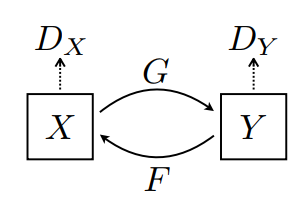


Figure 2 CycleGAN architecture from paper

The implementation of CycleGAN for this project will be described in chapter 3.

# Remove Makeup System

The system includes two main models. The first one takes responsibility for classifying if a face image has makeup or not. Based on that information, the system will decide whether it should remove makeup or not. The second model will remove makeup from the image. Finally, we created a user-friendly web application facilitating uploading images for makeup classification and removal.

## Dataset

**Folder Structure:**

* Makeup Images Folder: This folder contains images where individuals are wearing makeup. These images may include various styles of makeup, such as eye shadow, lipstick, foundation, etc.
* Non-Makeup Images Folder: This folder contains images where individuals are not wearing any makeup.

**Image Data:**

* Format: The images in both folders are likely to be in a specific format, such as JPEG, PNG, or another common image format.
* Resolution: Images may have varying resolutions.
* Name: Cycle GAN does not require paired image dataset; therefore, we don’t have any rules in naming images.

Dataset in this project take from different source (Thank to Mr. Vinh, we also use additional dataset from TikTok):

|  |  |  |  |
| --- | --- | --- | --- |
| **Dataset** | **Subjects** | **Images per subject** | **Total number of images** |
| **YMU** [7] | 151 | 4 (2 before and 2 after makeup application) | 604 |
| **VMU** [7] | 51 | 4 (1 no makeup, 1 lipstick, 1 eye makeup, 1 full makeover) | 204 |
| **MIW** [8] | 125 | 1-2 | 154 images (77 with makeup, 77 without makeup) |
| **MIFS** [9] | 107 subjects + 107 target subjects | 4 (2 before, 2 after makeup application) + 2 target | 642 |

Table 1 Dataset summary

## Classification Model

A classification model in deep learning is a type of artificial intelligence (AI) model designed to categorize input data into predefined classes or categories. The primary goal is to learn mapping from input features to a specific output class, enabling the model to make predictions or decisions on new, unseen data. Here are some important components implemented in PyTorch:

## 

Figure 3 Implement Classification Model

We use built-in models for this task. The loss function is Cross Entropy which is very popular in classification.

## Makeup Removal Model

The Interesting part of this project is makeup removal model. This model is based on CycleGAN. Here's a breakdown of the CycleGAN architecture and its key components (note: X is makeup image; Y is non makeup image):

* Generator (G\_XtoY and G\_YtoX): The Generator includes an encoder which compress image into latent vector and decoder which reconstruct image from latent vector. There are two generators in CycleGAN, denoted as G\_XtoY and G\_YtoX. G\_XtoY takes images from domain X and generates corresponding images in domain Y. G\_YtoX takes images from domain Y and generates corresponding images in domain X. Both generators are responsible for learning the mapping between the two domains.

A screen shot of a computer program

Description automatically generated

Figure 4 Generator Implementation

* Discriminator (D\_X and D\_Y): There are two discriminators, D\_X and D\_Y. D\_X evaluates the authenticity of images in domain X, distinguishing between real images from domain X and those generated by G\_YtoX. D\_Y evaluates the authenticity of images in domain Y, distinguishing between real images from domain Y and those generated by G\_XtoY.

A computer screen shot of a program code

Description automatically generated

Figure 5 Discriminator Implementation

Beside that we also have several loss functions:

* The generator loss is used to train the generators to produce realistic images, fooling the discriminators.
* The discriminator loss aims to correctly classify whether an image is real or generated, providing feedback to the generators to improve their performance.
* The key innovation in CycleGAN is the introduction of cycle-consistency loss. It enforces that if an image from domain X is translated to domain Y by G\_XtoT and then translated back to domain X by G\_YtoX, the result should be close to the original image. Similarly, if an image from domain Y is translated to domain X by G\_YtoX and then back to domain Y by G\_XtoY, the result should be close to the original image.
* The last loss function is identity loss, make sure that the generator creates a smooth image.

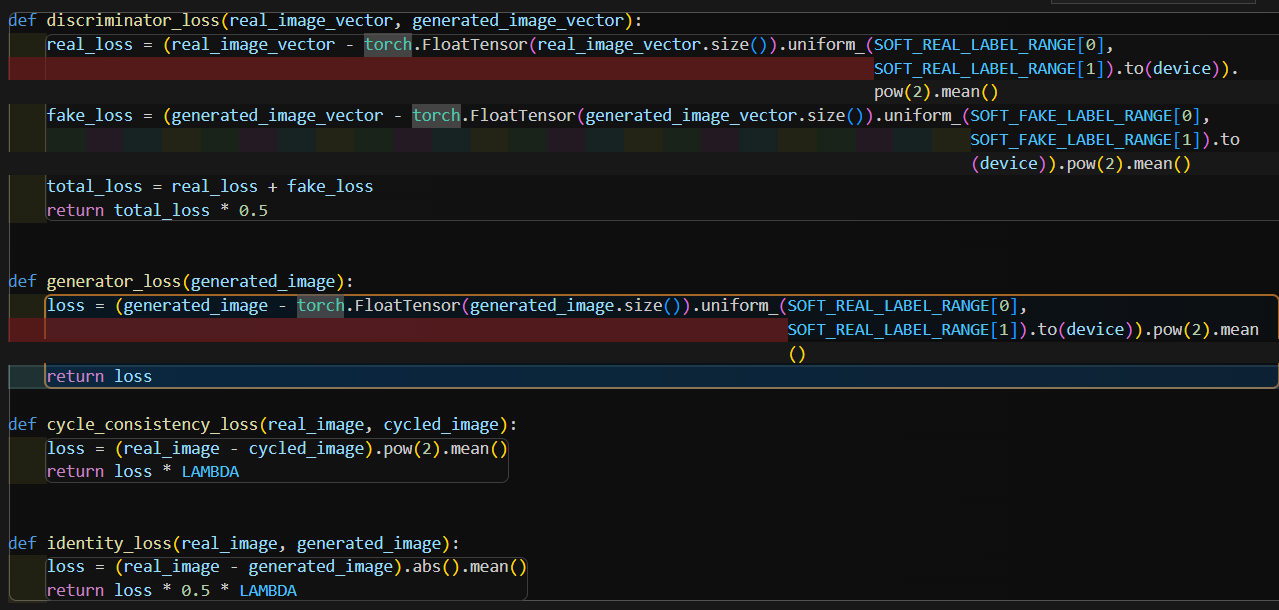


Figure 6 Loss Function Implementation

## Instruction for Use

You can try our product [here](https://anti-makeup.streamlit.app/) and [github](https://github.com/lam1101999/makeup-removal).

The workflow of our product is demonstrated as below:

A diagram of a computer system

Description automatically generated with medium confidence

When accessing our website, you can upload one or multiple images. One model will detect faces from your image and classify if that face has makeup or not. If that face has makeup, it will be removed makeup. Then the result is shown next to your original face.

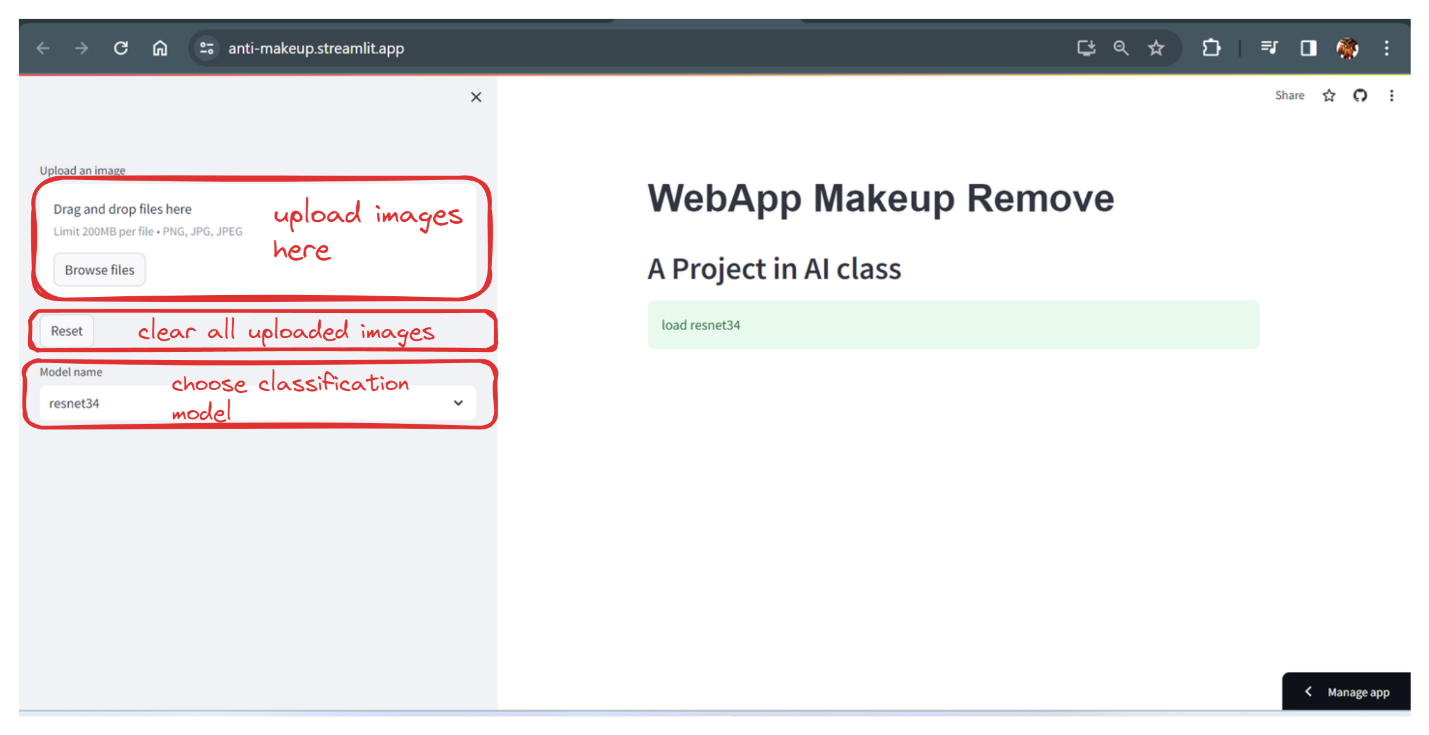


Figure 7 Basic functions

A screenshot of a web page

Description automatically generated

Figure 8 Example of product

# Result

In order to create classification model, we tried three different models: resnet34, mobilenet\_v3, efficientnet\_b1. The result of each model is recorded using tensorboard:

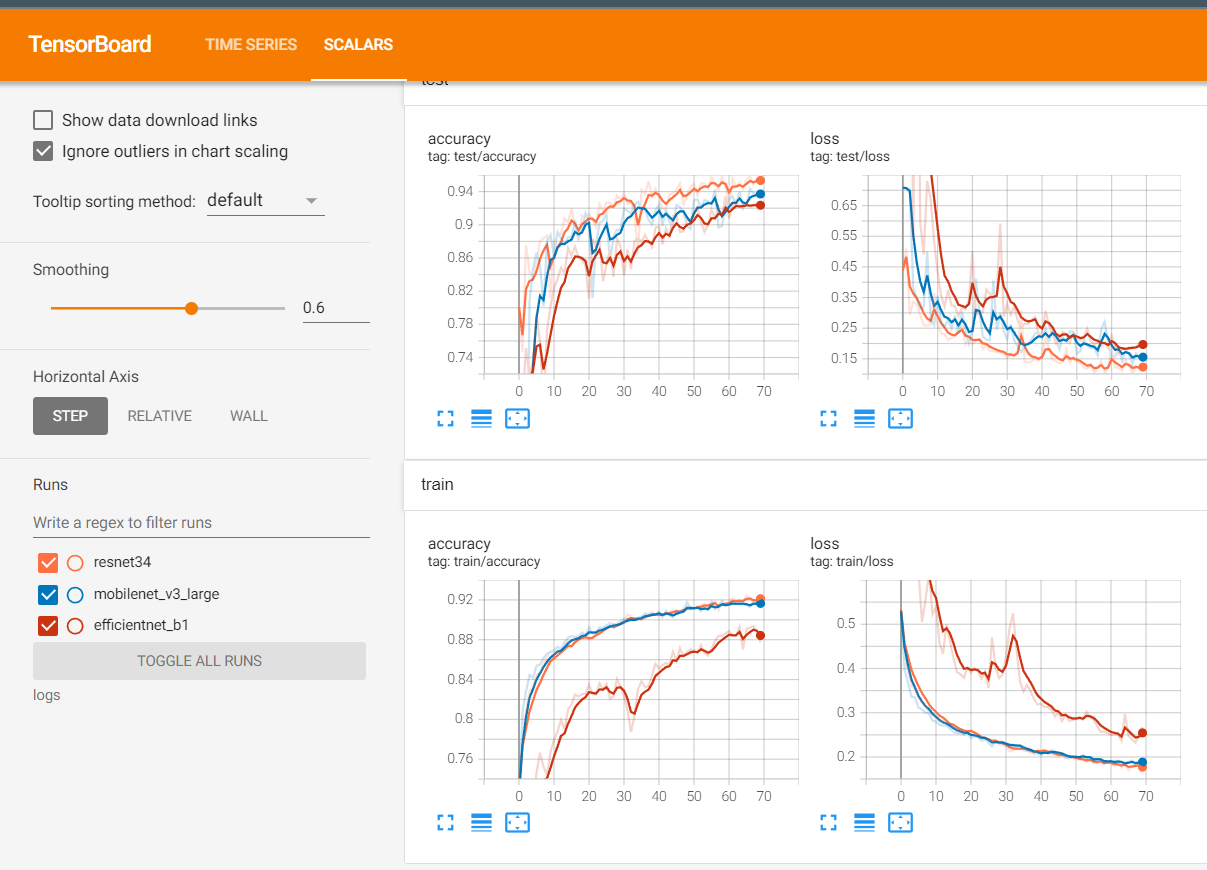


Figure 9 classification model result

The fact that all three models show good loss and accuracy indicates that our training process has been effective and reproducible. The ResNet-34 model outperforms the other models, implying that its architecture, which includes residual connections, is well-suited for the complexity of our dataset.

In the case of remove makeup model, the result is more complicated, when discriminator loss seems to be stable in both train and test phase. But the generator loss goes up and down a lot.

A screenshot of a graph

Description automatically generated

Figure 10 Gennerator loss

# References

|  |  |
| --- | --- |
| [1] | Leon A. Gatys, Alexander S. Ecker, Matthias Bethge, "A Neural Algorithm of Artistic Style". |
| [2] | Learning Enriched Features for Real Image Restoration and Enhancement, "Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, Ling Shao". |
| [3] | Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, Runmin Cong, "Zero-Reference Deep Curve Estimation for Low-Light Image Enhancement". |
| [4] | Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, "Generative Adversarial Networks". |
| [5] | Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks". |
| [6] | Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks". |
| [7] | A. Dantcheva, C. Chen, A. Ross, "Can Facial Cosmetics Affect the Matching Accuracy of Face Recognition Systems?," in *5th IEEE International Conference on Biometrics: Theory, Applications and Systems*. |
| [8] | C. Chen, A. Dantcheva, A. Ross, "Automatic Facial Makeup Detection with Application in Face Recognition," 6th IAPR International Conference on Biometrics. |
| [9] | C. Chen, A. Dantcheva, T. Swearingen, A. Ross, "Spoofing Faces Using Makeup: An Investigative Study," in *3rd IEEE International Conference on Identity, Security and Behavior Analysis*. |